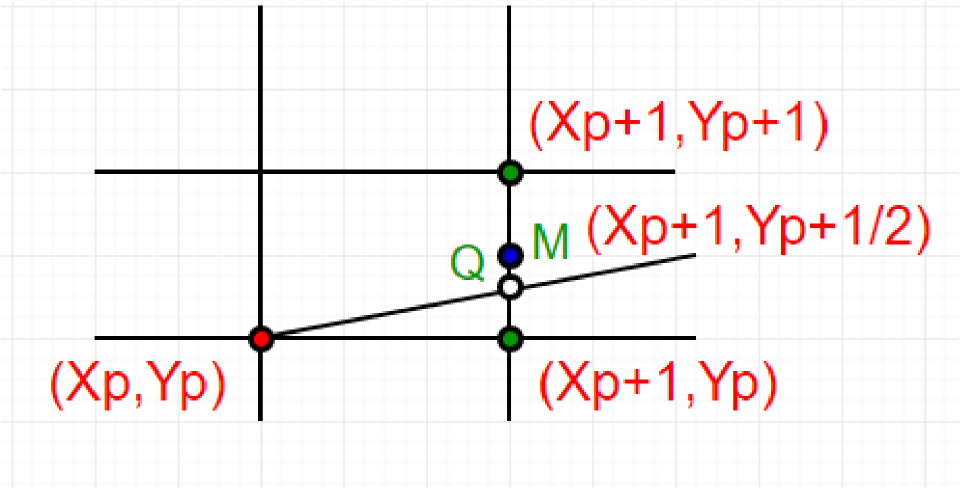
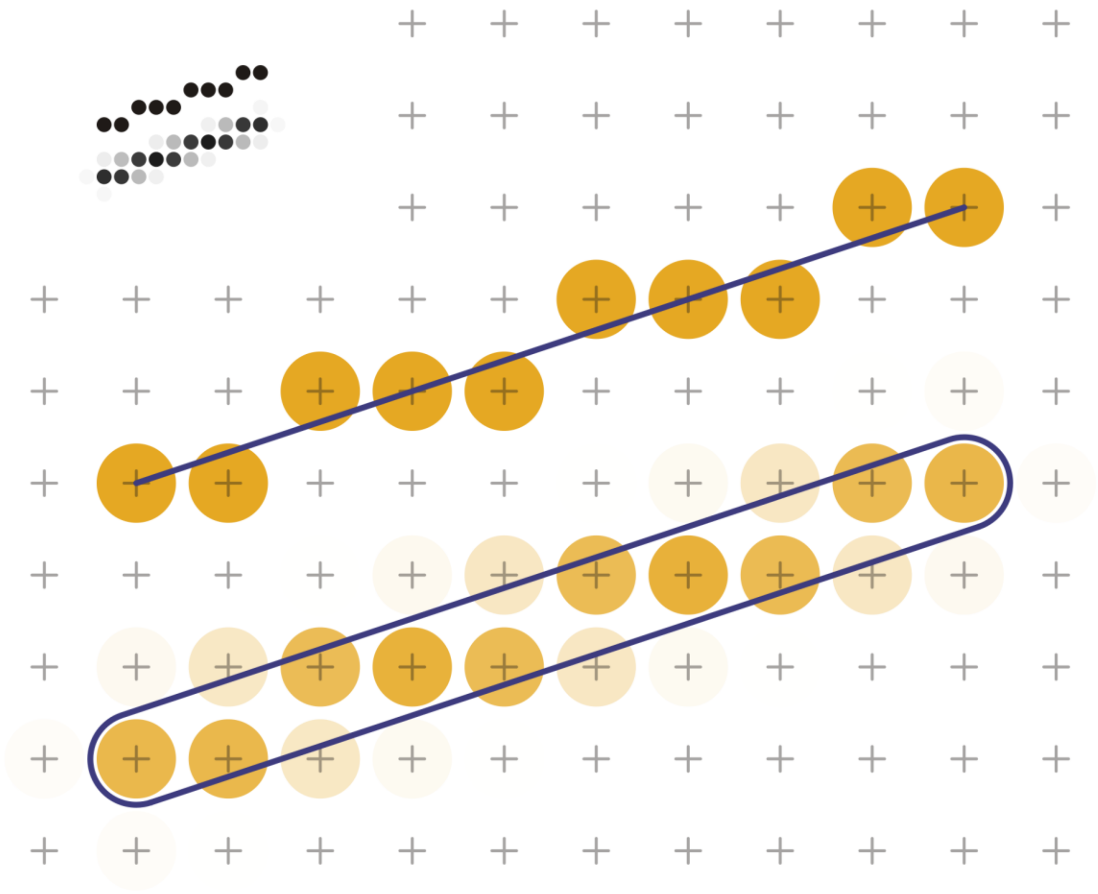
За вътрешно ползване!
ТУЕС

# Алгоритъм за бърза растеризация на линия Brsenham / Mid-point Algorithm

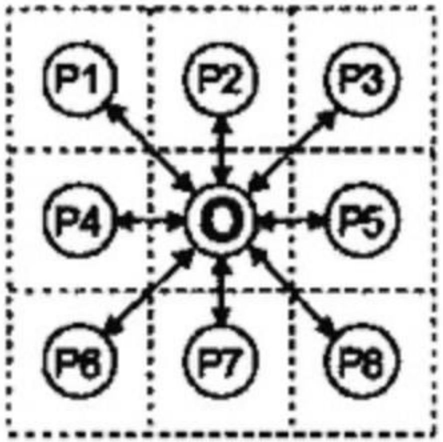
д-р инж. Росен Петков



# Растеризация на линия



$y = kx + b$



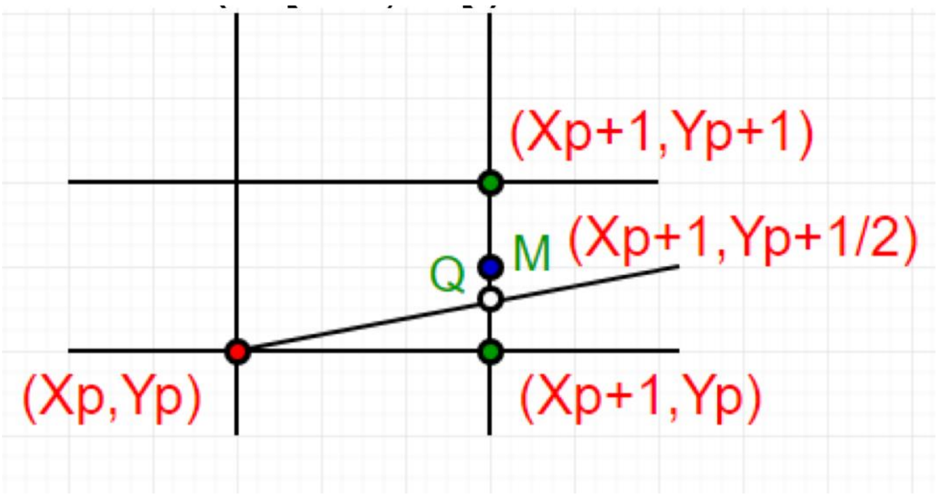
Дадена е координата на две точки A(x1, y1) и B(x2, y2), така че x1 < x2 и y1 < y2. Задачата да се намерят всички междинни точки, необходими за начертаване на линия AB на екрана на компютъра от пиксели. Имайте предвид, че всеки пиксел има цяло число координати.

Наклонът на линията (к) е между 0 и 1, т.е. <=45 градуса. Начертаваме линия от долния ляв към горния десен ъгъл. Ако това не е така, сменят се осите x и y и се изпълнява условието.

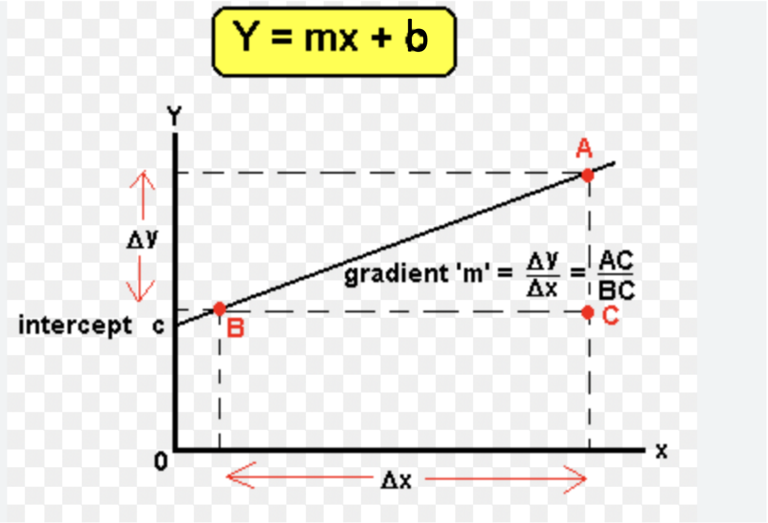
За всеки даден/изчислен предходен пиксел P(Xp,Yp), има два кандидата за следващия пиксел най-близо до линията, E(Xp+1, Yp) и NE(Xp+1, Yp+1) (E означава Изток и СИ означава североизток).

В алгоритъма на Mid-Point правим следното.
Намераме средата на две възможни следващи точки. Средата на E(Xp+1, Yp) и NE(Xp+1, Yp+1) е M(Xp+1, Yp+1/2).

Ако M е над линията, изберете E като следваща точка.
Ако M е под линията, изберете NE като следваща точка.



Нека разгледаме линия y = mx + B.




Можем да пренапишем уравнението като:
y = (dy/dx)x + B или
(dy)x + B(dx) - y(dx) = 0

Нека F(x, y) = (dy)x - y(dx) + B(dx)

-> За всички точки (x,y) на линията, решението на F(x, y) е 0.

-> За всички точки (x,y) над линията, F(x, y) води до отрицателно число.
-> И за всички точки (x,y) под линията, F(x, y) води до положително число.

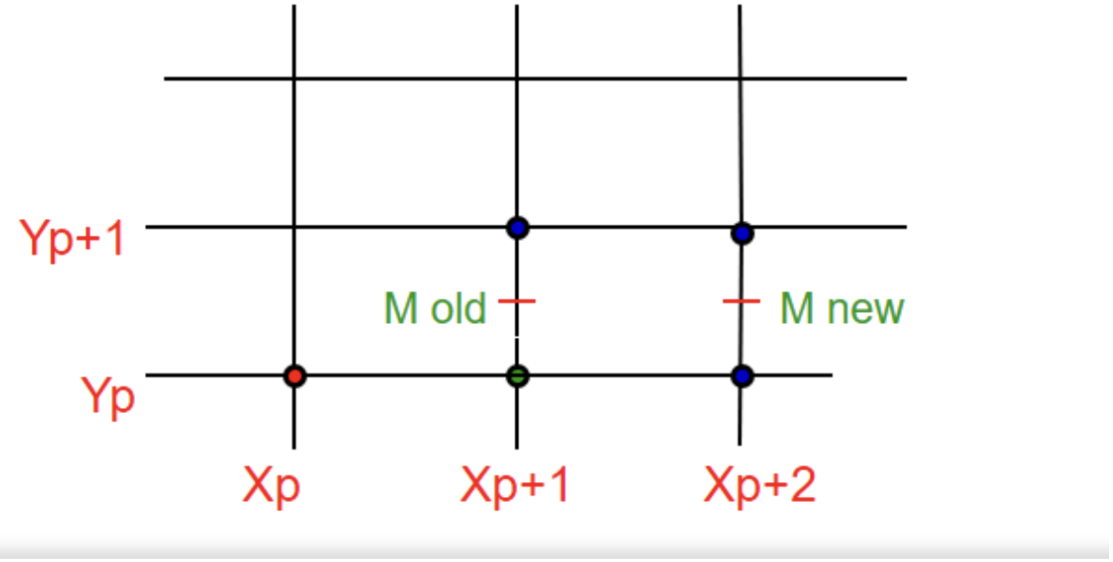
Тази връзка се използва за определяне на относителната позицията на M

M = (Xp+1, Yp+1/2)
Така че нашият параметър за решение d е,
d = F(M) = F(Xp+1, Yp+1/2)

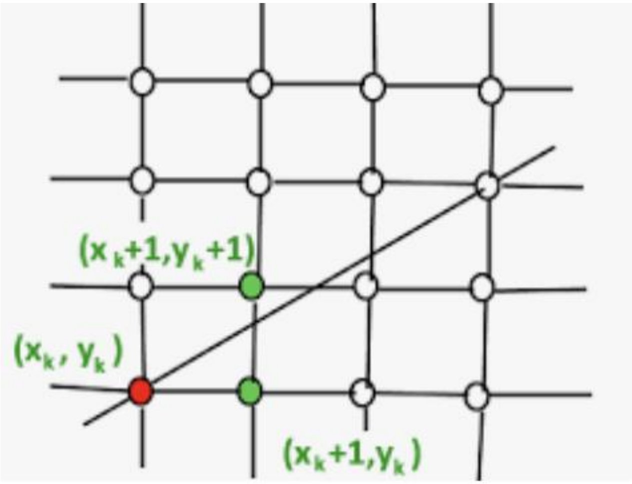
Как ефективно да се намери нова стойност на d от старата му стойност?
За простота нека запишем F(x, y) като ax + by + c.
Където a = dy
b = -dx
c = B*dx

**Case 1:** Ако е избрана E за следваща точка :
dnew = $F(X_p+2, Y_{p+1}/2)$
$= a(X_p+2) + b(Y_{p+1}/2) + c$
dold = $a(X_p+1) + b(Y_{p+1}/2) + c$
Разликата:
Deld = dnew – dold
$= a(X_p+2)- a(X_p+1)+ b(Y_{p+1}/2)- b(Y_{p+1}/2)+ c-c$
$= a(X_p) +2a – a(X_p) – a$
$= a.$

т.е dnew = dold + dy. (a = dy)



**Case 2:** Ако е избрана NE за следваща точка:
dnew = $F(X_p+2, Y_p+3/2)$
$= a(X_p+2) + b(Y_p+3/2) + c$
dold = $a(X_p+1) + b(Y_p+1/2) + c$




Делта:
Deld = dnew -dold
$= a(X_p+2)- a(X_p+1)+ b(Y_p+3/2)- b(Y_p+1/2)+ c-c$
$= a(X_p) + 2a – a(X_p) – a + b(Y_p) + 3/2b – b(Y_p) -1/2b$
$= a + b$

т.е, dnew = dold + dy – dx. (a = dy , b = -dx)

**Първоначалната стойност на d0:**

$d0 = F(X1+1 , Y1+1/2)$
$= a(X1 + 1) + b(Y1 + 1/2) +c$
$= aX1+ bY1 + c + a + b/2$
$= F(X1,Y1) + a + b/2$
$= a + b/2 \quad (F(X1, Y1) = 0 )$

$d0 = dy – dx/2. \ (a = dy, b = -dx)$

# Алгоритъм

```
Input (X1,Y1) and (X2,Y2)
dy = Y2− Y1
dx = X2 − X1
// initial value of
// decision parameter d


if(dy<=dx){
d = dy − (dx/2)
x = X1 , y = Y1

// plot initial given point
Plot(x , y)

// iterate through value of X
while(x < X2)
    x = x+1

    // 'E' is chosen
    if (d < 0)
       d = d + dy
```

```
    // 'NE' is chosen
    else
       d = d + dy − dx
       y = y+1
    Plot(x,y)}

else if(dx<=dy)
{
d = dx − (dy/2)
x = X1 , y = Y1

// plot initial given point
Plot(x , y)

// iterate through value of X
while(y< Y2)
    y= y+1

    // 'E' is chosen
    if (d < 0)
       d = d + dx

    // 'NE' is chosen
    else
       d = d + dx − dy
       x= x+1
    Plot(x,y)
}
```